

Priprema za izvođenje nastave – ogledni primjer prilagodbi metoda, sadržaja i aktivnosti za učenike s teškoćama

Škola:	Strukovna škola Đurđevac
Mjesto:	Đurđevac
Obrazovni sektor:	Elektrotehnika i računalstvo
Strukovna kvalifikacija:	Tehničar za računalstvo
Odjel:	1b
Nastavni predmet:	Algoritmi i programiranje
Naziv teme:	Naredba break
Nastavnik:	Tihomir Robotić

Napomena: prilagodba sadržaja, metoda i zahtjeva ovisi o specifičnim teškoćama koje učenik ima. U ovom radu fokus je na spoznajnoj prilagodbi i prilagodbi školskih zahtjeva.

Planirani tijek aktivnosti

Vremensko trajanje aktivnosti	2 školska sata
Odgojno obrazovni ishodi	Primijeniti naredbu break za ispravno prekidanje petlji Rješenje zapisati u programskom jeziku C#
Međupredmetne teme	Učiti kako učiti A.4/5.2. Učenik se koristi različitim strategijama učenja i samostalno ih primjenjuje u ostvarivanju ciljeva učenja i rješavanju problema u svim područjima učenja. B.4/5.4. Učenik samovrednuje proces učenja i svoje rezultate, procjenjuje ostvareni napredak te na temelju toga planira buduće učenje. Osobni i socijalni razvoj A.4.3. Razvija osobne potencijale. A.4.4. Upravlja svojim obrazovnim i profesionalnim putem.
Opis aktivnosti	Uvodni dio Učenik ponavlja petlje for, while i do while Očekivano predznanje – učenik primjenjuje petlje for, while i do while na ispravan način, učenik razlikuje tipove zadataka s ponavljanjem poznati broj puta i tipove zadataka s ponavljanjem nepoznat broj puta Pripremni dio Motivacijsko pitanje – može li se petlja koja je započela svoj rad kontrolirano prekinuti? Glavni dio – aktivnost 1 Nastavnik koristi metodu demonstracije U primjeru 1 demonstriramo prekidanje for petlje naredbom break U primjeru 2 demonstriramo prekidanje while petlje naredbom break U primjeru 3 demonstriramo prekidanje do while petlje naredbom break Vježbanje – aktivnost 2 U ovoj aktivnosti učenici samostalno rješavaju problem Potrebno je napisati program koji od korisnika traži unos broja n. Nakon toga potrebno je unijeti n brojeva koje treba pribrojiti. Prekinuti program ako korisnik dva put za redom unese isti broj. Ponovljeni broj ne pribrajati. Program sadrži: Unos i ispis podataka Aritmetičke operacije For petlju If grananje Naredbu break Vrednovanje Učenik dobiva povratnu informaciju o usvojenosti novih sadržaja Nastavnik dobiva povratnu informaciju o stupnju usvojenosti sadržaja od strane učenika

	Poticanje natjecateljskog duha rješavanjem zadataka i uspoređivanje svojih odgovora s odgovorima drugih učenika Vrednovanje za učenje: Nastavnik prati aktivnosti tijekom rada Učenik kroz razgovor s nastavnikom dobiva povratnu informaciju o svom načinu razmišljanja i tehnikama rješavanja problema Nastavnik kroz odgovore učenika dobiva povratnu informaciju u kojoj mjeri je njegova aktivnost razumljiva te jesu li aktivnosti za učenike razumljivo postavljene Vrednovanje kao učenje: Učenici samovrednuju svoje odgovore Učenici vrednuju odgovore drugih učenika
--	---

Organizacija rada – artikulacija aktivnosti

Nastavna sredstva, pomagala i ostali materijalni uvjeti:

- Informatički praktikum opremljen računalima, projektorom i platnom za prikaz
- Integrirana razvojna okolina Microsoft Visual Studio Community 2022
- Softverski okvir .NET 7.0
- Programski jezik C#

Napomena: Radi pojednostavljenja izvornog koda, koristimo Top Level Statements (TLS)

Nastavne metode:

- Usmeno izlaganje
- Razgovor
- Postavljanje pitanja
- Praktičan rad
- Demonstracija
- Samostalno rješavanje zadataka
- Rješavanje problema

Metodički oblici rada:

- Individualni rad
- Frontalni rad

Metode za spoznajno prilagođavanje

- Isticanje bitnoga različitim tipovima obilježavanja i/ili slikovnog predočavanja
- Raščlanjivanje složenih zadataka
- Semantičko pojednostavljivanje sadržaja učenja preradom sadržaja u smislu uporabe jednostavnog govornog izraza

Metode za prilagodbu školskih zahtjeva

- Doziranje zadataka, postepeno uz poticanje samostalnosti u radu
- Mogućnost duljeg vremena za rad

Tijek izvođenja nastave

1. Uvodni dio

Učenik ponavlja petlje for i while

Očekivano predznanje – učenik primjenjuje petlje for, while i do while na ispravan način, učenik razlikuje tipove zadataka s ponavljanjem poznati broj puta i tipove zadataka s ponavljanjem nepoznat broj puta

<pre>int zbroj = 0; int koraci = 0; for (int i = 1; i < 5; i++) { zbroj += i; koraci++; } Console.WriteLine(zbroj); Console.WriteLine(koraci);</pre>	Očekivani izlaz: 10 4
<pre>int broj = 1389; int koraci = 0; while(broj > 0) { broj /= 10; koraci++; } Console.WriteLine(koraci);</pre>	Očekivani izlaz: 4
<pre>int broj = 1; int koraci = 0; do { broj += 10; koraci++; } while (broj < 100); Console.WriteLine(koraci);</pre>	Očekivani izlaz: 10

2. Pripremni dio

Motivacijsko pitanje – može li se petlja koja je započela svoj rad kontrolirano prekinuti?

Najava glavnog dijela – u nastavku ćemo demonstrirati slučajeve kada je potrebno prekinuti petlje prije nego što su završile svoj rad do kraja

3. Glavni dio

Aktivnost 1 – Nastavnik koristi metodu demonstracije

U primjeru 1 demonstriramo prekidanje for petlje naredbom break

U primjeru 2 demonstriramo prekidanje while petlje naredbom break

U primjeru 3 demonstriramo prekidanje do while petlje naredbom break

Primjer 1 – demonstriramo prekidanje for petlje naredbom break. Od korisnika tražimo unos 10 pozitivnih cijelih brojeva koje treba zbrojiti. Ako korisnik unese negativan broj, petlja se prekida u tom koraku u kojem je negativni broj unesen i ispisuje se rezultat zbrojenih brojeva do tog koraka.

```
int broj = 0;
int zbroj = 0;
Console.WriteLine("Unesi 10 pozitivnih brojeva koji će se zbrojiti!");
for (int i = 0; i < 10; i++)
{
    Console.Write($"Unesi {i + 1}. broj --> ");
    broj = Convert.ToInt32(Console.ReadLine());
    if (broj < 0)
    {
        Console.WriteLine("Unio si negativan broj!");
        break;
    }
    zbroj += broj;
}
Console.WriteLine($"Zbroj brojeva: {zbroj}");
Console.ReadKey();

Console.ReadKey();
```

Očekivani izlaz:

```
Unesi 10 pozitivnih brojeva koji će se zbrojiti!
Unesi 1. broj --> 1
Unesi 2. broj --> 2
Unesi 3. broj --> 3
Unesi 4. broj --> 4
Unesi 5. broj --> 5
Unesi 6. broj --> -10
Unio si negativan broj!
Zbroj brojeva: 15
```

Primjer 2 – demonstriramo prekidanje while petlje naredbom break. Od korisnika tražimo unos pozitivnih cijelih brojeva u rasponu od 1 do 1000. Uneseni brojevi se pribrajaju. Ako se u bilo kojem koraku unese broj 17, petlju prekinuti i ispisati trenutni zbroj.

```
int zbroj = 0;
int broj = 1;
Console.WriteLine("Unosi brojeve između 1 i 1000 OSIM broja 17! Brojevi se pribrajaju!");
while (broj >= 1 && broj <= 1000)
{
    Console.Write("Unesi broj --> ");
    broj = Convert.ToInt32(Console.ReadLine());
    if (broj == 17)
    {
        Console.WriteLine("Unio si broj 17!");
        break;
    }
}
```

```

else if (broj >= 1 && broj <= 1000)
{
    zbroj += broj;
}
}
Console.WriteLine($"Zbroj unesenih brojeva: {zbroj}");
Console.ReadKey();

```

Očekivani izlaz:

Unosi brojeve između 1 i 1000 OSIM broja 17!
 Brojevi se pribrajaju!
 Unesi broj --> 1
 Unesi broj --> 49
 Unesi broj --> 100
 Unesi broj --> 1003
 Zbroj unesenih brojeva: 150

Unosi brojeve između 1 i 1000 OSIM broja 17!
 Brojevi se pribrajaju!
 Unesi broj --> 10
 Unesi broj --> 56
 Unesi broj --> 17
 Unio si broj 17!
 Zbroj unesenih brojeva: 66

Primjer 3 – demonstriramo prekidanje do while petlje naredbom break. Od korisnika se traži unos pozitivnih cijelih brojeva sve dok ukupan zbroj ne dosegne vrijednost 1000. Ako se u bilo kojem trenutku unese negativan broj, program se prekida. Ispisati koliko brojeva je uneseno prije regularnog završetka ili prekida programa.

```

int zbroj = 0;
int broj = 1;
int brojeva = 0;
Console.WriteLine("Unosi brojeve! Brojevi se pribrajaju sve do zbroj 1000!");
do
{
    Console.Write("Unesi broj --> ");
    broj = Convert.ToInt32(Console.ReadLine());
    if (broj < 0)
    {
        Console.WriteLine("Unio si negativan broj!");
        break;
    }
    else if (broj >= 1 && broj <= 1000)
    {
        zbroj += broj;
        brojeva++;
    }
} while (zbroj < 1000);
Console.WriteLine($"Unesenih brojeva: {brojeva}");
Console.ReadKey();

```

Očekivani izlaz:

Unosi brojeve! Brojevi se pribrajaju sve do zbroj 1000!
 Unesi broj --> 1
 Unesi broj --> 500
 Unesi broj --> 678
 Unesenih brojeva: 3

Unosi brojeve! Brojevi se pribrajaju sve do zbroj 1000!
 Unesi broj --> 1
 Unesi broj --> 2
 Unesi broj --> 3
 Unesi broj --> -7
 Unio si negativan broj!
 Unesenih brojeva: 3

Aktivnost 2 – u ovoj aktivnosti učenici samostalno rješavaju problem

Potrebno je napisati program koji od korisnika traži unos broja *n*. Nakon toga potrebno je unijeti *n* brojeva koje treba pribrojiti. Prekinuti program ako korisnik dva puta za redom unese isti broj. Ponovljeni broj ne pribrajati.

Očekivani izlaz:

Unesi broj n --> 5	Unesi broj n --> 10
Unesi 1. broj --> 1	Unesi 1. broj --> 1
Unesi 2. broj --> 2	Unesi 2. broj --> 2
Unesi 3. broj --> 3	Unesi 3. broj --> 3
Unesi 4. broj --> 4	Unesi 4. broj --> 3
Unesi 5. broj --> 5	Unijeli ste dva puta za redom isti broj!
Zbroj brojeva: 15	Zbroj brojeva: 6

Metode za spoznajnu prilagodbu

1. Motivirati učenika na sastavljanje liste varijabli koje ćemo koristiti u programu. Također, motivirati učenika na promišljanje oko tipova podataka koje treba pridijeliti svakoj varijabli.

Primjer:

Varijabla	Tip podatka
N	int
broj	int
zbroj	int
prethodni	int

2. Motivirati učenika na ispravno formatiranje poruka koje upućujemo korisniku prilikom unosa podataka i na ispravno pridjeljivanje unesene vrijednosti odgovarajućoj varijabli.

```
Console.WriteLine("Unesi broj n --> ");  
n = Convert.ToInt32(Console.ReadLine());
```

3. Motivirati učenika na sastavljanje ispravne petlje. U ovom primjeru, najispravnije je kreirati for petlju koja prolazi kroz raspon brojeva od 1 do *n*, uključujući *n*.

```
for (int i = 1; i <= n; i++)  
{  
  
}
```

4. Motivirati učenika na ispravno formatiranje poruke koju upućujemo korisniku prilikom unosa broja unutar for petlje i na ispravno pridjeljivanje unesene vrijednosti odgovarajućoj varijabli.

```
Console.WriteLine($"Unesi {i}. broj --> ");  
broj = Convert.ToInt32(Console.ReadLine());
```

5. Motivirati učenika na provjeru je li uneseni broj jednak prethodnom unesenom broju. Također, potrebno je korisniku ispisati prikladnu poruku uz objašnjenje zašto se program prekida.

```
if (broj == prethodni)  
{  
    Console.WriteLine("Unijeli ste dva puta za redom isti broj!");  
    break;  
}  
  
prethodni = broj;
```

6. Motivirati učenika na pribrajanje unesenih brojeva u varijablu zbroj

```
zbroj += broj;
```

7. Motivirati učenika na ispis ukupnog zbroja unesenih brojeva po završetku programa

```
Console.WriteLine($"Zbroj brojeva: {zbroj}");
```

8. Raščlanite zadatak na manje cjeline

Cjelina 1 – deklaracija varijabli

```
int n = 0;
int broj = 0;
int zbroj = 0;
int prethodni = 0;
```

Cjelina 2 – unos podatka koji nam govori koliko brojeva unosimo

```
Console.Write("Unesi broj n --> ");
n = Convert.ToInt32(Console.ReadLine());
```

Cjelina 3 – kreiranje petlje za unos n brojeva

```
for (int i = 1; i <= n; i++)
{
}
}
```

Cjelina 4 – unos podatka – unosimo broj unutar tijela petlje

```
Console.WriteLine($"Unesi {i}. broj --> ");
broj = Convert.ToInt32(Console.ReadLine());
```

Cjelina 5 – provjera je li uneseni broj jednak prethodnom, prekid programa ako je odgovor da

```
if (broj == prethodni)
{
    Console.WriteLine("Unijeli ste dva puta za redom isti broj!");
    break;
}
```

```
prethodni = broj;
```

Cjelina 6 – pribrajanje unesenih brojeva

```
zbroj += broj;
```

Cjelina 7 – ispis ukupnog zbroja unesenih brojeva

```
Console.WriteLine($"Zbroj brojeva: {zbroj}");
```

Metode za prilagodbu školskih zahtjeva:

1. Usmjerite učenika na samostalno rješavanje zadatka postupno , po cjelinama.
Primjer:

Cjelina 1 – deklariraj sve potrebne varijable
Cjelina 2 – unesi broj koji nam govori koliko brojeva unosimo
Cjelina 3 – kreiraj petlju za unos brojeva
Cjelina 4 – unutar petlje unosi novi broj u svakom koraku petlje
Cjelina 5 – provjeri je li uneseni broj jednak prethodnom unesenom broju. Ako je odgovor da, prekini program i prikaži korisniku pripadnu poruku
Cjelina 6 – pribraja unesene brojeve u posebnu varijablu
Cjelina 7 – po završetku program ispiši ukupan zbroj unesenih brojeva
2. Učenik rješava zadatak samostalno po cjelinama, Nakon svake cjeline dajte učeniku povratnu informaciju i usmjerite ga prema slijedećoj cjelini
3. Omogućite učeniku dulje vrijeme za rješavanje zadatka

Očekivano rješenje:

```
int n = 0;
int broj = 0;
int zbroj = 0;
int prethodni = 0;

Console.WriteLine("Unesi broj n --> ");
n = Convert.ToInt32(Console.ReadLine());

for (int i = 1; i <= n; i++)
{
    Console.WriteLine($"Unesi {i}. broj --> ");
    broj = Convert.ToInt32(Console.ReadLine());
    if (broj == prethodni)
    {
        Console.WriteLine("Unijeli ste dva puta za redom isti broj!");
        break;
    }

    prethodni = broj;
    zbroj += broj;
}
Console.WriteLine($"Zbroj brojeva: {zbroj}");
Console.ReadKey();
```

Vrednovanje

Učenik dobiva povratnu informaciju o usvojenosti novih sadržaja

Nastavnik dobiva povratnu informaciju o stupnju usvojenosti sadržaja od strane učenika

Poticanje natjecateljskog duha rješavanjem zadataka i uspoređivanje svojih odgovora s odgovorima drugih učenika

Vrednovanje za učenje:

Nastavnik prati aktivnosti tijekom rada

Učenik kroz razgovor s nastavnikom dobiva povratnu informaciju o svom načinu razmišljanja i tehnikama rješavanja problema

Nastavnik kroz odgovore učenika dobiva povratnu informaciju u kojoj mjeri je njegova aktivnost razumljiva te jesu li aktivnosti za učenike razumljivo postavljene

Vrednovanje kao učenje:

Učenici samovrednuju svoje odgovore

Učenici vrednuju odgovore drugih učenika

Evaluacija nastavnog sata

Poželjno je izraditi ili anketni listić ili online anketu u kojoj nastavnik dobiva anonimnu povratnu informaciju od učenika